

Enhancing Economics and Finance Learning through Automated Grading of Spreadsheet Exercises

Douglas M. Kline¹ and Thomas N. Janicki²

Abstract

Instructors of undergraduate courses in economics and finance as well as other business school courses are increasingly turning to spreadsheet applications to reinforce concepts presented in the classroom. As the number of exercises increases in these courses the grading and management of the assignments become an overwhelming burden. This teaching note discusses five techniques that were implemented to increase the number of learning exercises, while decreasing the instructors grading time and the opportunity for plagiarism. The key success factor was to implement an automatic grading program for Microsoft Office products reducing grading timing and plagiarism while increasing objectivity in the grading.

Introduction

Instructors of economics and finance courses as well as other undergraduate business courses are finding that students need an increasing level of computer-based personal productivity software skills in order to successfully complete their content coursework. Personal productivity software applications are defined as applications found in a typical “office suite” of software products (word processing, spreadsheet, database and presentation software). Many undergraduate courses employ the use of spreadsheet software (prior to introducing upper level courses’ customized application software) to reinforce concepts in the classroom and bring the concepts to life.

To effectively learn new economic and financial concepts, students complete hands-on assignments, and receive feedback for these assignments. However, as instructors increase the number of software exercises in these courses, they encounter problems such as:

- Increased time required to grade exercises
- Less ability to customize feedback
- Delays in providing feedback
- Increased plagiarism

Because of the above problems, instructors must weigh the learning benefit of adding additional required homework exercises against their time constraints. Many times the decisions made compromise the ultimate goal of student learning.

This teaching note paper has three areas of emphasis. First an overview of learning theory will be discussed which lays the foundation for the need of multiple exercises in undergraduate courses. This will be followed by an overview of an approach for the automatic grading of spreadsheet applications implemented in the same undergraduate course over the past three years by two different instructors (total of six sections). Limited demographics and summary student course evaluations are provided, as well as plans for the future. Finally a few observations and results from this implementation will be discussed.

¹ Associate Professor of Information Systems, Cameron School of Business, University of North Carolina at Wilmington, 601 S. College Rd., Wilmington, NC 28403-5611, email: klined@uncw.edu

² Assistant Professor of Information Systems, Cameron School of Business, University of North Carolina at Wilmington, 601 S. College Rd., Wilmington, NC 28403-5611, email: janickit@uncw.edu

Background

One instructional learning style builds from a theoretical foundation known as behavior psychology learning style. Behavioral psychology (also known as Skinnerian or stimulus response psychology) is based on the premise that learning results from the pairing of *response* with *stimuli* (Hannafin & Peck, 1988.) In this view, a quick response to a stimulus helps the individual to learn.

Building on this behavioral learning theory, Gagne, Briggs & Wager (1988) propose several design principles for effective instructional design. These principles are **contiguity**, **repetition**, and **feedback**.

Contiguity is the concept that the response should immediately follow the stimulus. Lagging responses decrease the likelihood of correct answers to similar questions in the future. Immediate response increases the likelihood of correct answers in the future. In the undergraduate curriculum context, submitting a spreadsheet to be graded, and receiving a grade several days later is a significant delay in the response.

The second concept, repetition, asserts that practice strengthens learning and improves retention. A style based on rote learning and memorization is encouraged by Murray (1996). In computer literacy courses, increased repetition takes the form of a greater number of assignments, or assignments with many tasks. However, with large sections and a large number of assignments, grading can be an overwhelming task.

The third concept, feedback, requires that students know whether their answer is correct or incorrect. Feedback that the answer is correct is as important as feedback that the answer is incorrect. Providing proper feedback increases the likelihood of learning (Anderson & Reiser, 1985). Furthermore, *immediate* feedback was also found by Saljo (1979) to promote learning in the areas of memorization, skill and fact acquisition, and interpreting and understanding reality. Typical computer literacy assignments have many parts that must be graded, and marked as correct or incorrect. To be useful, the feedback must identify the item being marked, whether it is correct or not, and ideally, refer the student to remedial content. As mentioned above, this can be onerous with a large number of assignments, sections, and students per section.

To address the above learning theories and concepts, the author's desired to increase the number of software application hands-on exercises into their undergraduate courses to reinforce course concepts. As the number of exercise increased so did new problems for the instructors: Included were:

- The physical handling of the submission of assignments became difficult because they can arrive in many forms, none of which are ideal. Paper submission of reports lends itself too easily to plagiarism, and do not allow graders to know what's actually in the file. Floppy disks take a great deal of time to swap in and out of the floppy drive as well as the physical handling of received diskettes and then getting them returned to the proper student. Email attachments can also be slow to process, and certain file formats can be blocked for security reasons (Microsoft Access *.mdb files are blocked by default in the recent MS Outlook security patch). Furthermore, a variety of file formats can greatly complicate grading.
- A second practical consideration is plagiarism. Assignments are generally file-based, and with increased connectivity and email access, plagiarism has become extremely convenient, and widespread. Typical spreadsheet assignments frequently require exactly the same thing from every student, so visual checking for plagiarism is ineffective.
- Finally, the instructor's time increases in order to grade multiple assignments and provide customized feedback quickly to learners.

Implementation of an Automated Grading System

As a result of these problems the authors began a process and build automation into the homework cycle to overcome some of the problems mentioned in the prior section. This section highlights some of the solutions that we have implemented to achieve higher quality learning of basic course concepts, while keeping the workload reasonable for the instructor.

Password protected class web site for the download of exercises

Students are assigned a username and password at the beginning of the semester for access to student-specific areas on our web site. Each student obtains their assignment from this password-protected area. As

an example, for a MS Excel assignment, a student would log in to the web site, and download a spreadsheet file. The student then edits the file as required by the assignment. This permitted the authors to populate a spreadsheet with raw data that the student then adds formulas and equations to the spreadsheet.

The download of these populated files helps in clarifying the assignment, and in controlling the format of the file. Clarification of the assignment is aided because instructions can be directly embedded in the document. In a spreadsheet assignment, cells that should have formulas can be highlighted. Of course, this can be adjusted as needed from completely blank documents to highly structured instructions. The format of the file is also initially set. So if every file is initially MS Excel 2002, more of the files handed in will be of that same version.

Creation of customized assignments

One feature of the download process is that all students need not receive the same file. These files are generated as the student downloads the files, so the instructor has the option to populate the spreadsheet with different data sets for each student. This is one of the manners employed to reduce plagiarism.

Embedding of student ID's into the downloaded files – Assists in detection of plagiarism

Another aspect of the downloading of assignments is that each student receives a file that contains an embedded identifier. All of the MS (Microsoft) Office products have document property fields that can be password protected. By embedding student identifiers in the files that students download, we can positively identify plagiarism (when the file is subsequently uploaded) and also identify all parties involved (who provided to file to be copied).

Plagiarism is an issue that has troubling ethical implications. From experience, we know that instructors can spend a relatively large amount off their time addressing this issue for a relatively few number of offenders. Furthermore, it can be extremely difficult to prove. As a result, many instructors do not address it in any way. Our perspective is that increased vigilance will result in increased compliance with the assignments, thus promoting repetition, and ultimately better learning for more students. For one author, this feature of embedding ID's into downloaded files provided evidence of plagiarism from twelve, eight and five students in three consecutive semesters who submitted another student's file as their own work. It is felt that the number of students involved in plagiarism dropped each semester as students spread the word that the computer was 'somehow' checking for duplicate files.

Online submission of exercises

When students have completed their assignments, they submit them in the same password protected web site where they received their assignment. The assignments are placed in a secure folder that the professor has access to. The file extension is checked to make sure the file that is uploaded is of the proper type for the assignment. For instance, a MS Excel spreadsheet assignment will check to make sure the file being uploaded has a ".xls" file extension. The files are time-stamped as they are uploaded.

From the student perspective, they have a single secure area where they can obtain and submit assignments from any web-enabled location, at any time. From the instructor perspective, assignments are in a single folder with timestamps to check for tardiness. This solved the problem of email attachments, diskette, zip drive handling and corrupted files.

Automated grading

We currently have implemented a custom Visual Basic program that grades MS Excel, files based on grading templates. The grading templates are text files that list items to be graded, allowable correct answers, the points per item, and the feedback comment for incorrect items.

Besides the automated grading of formulas and values, items can also be displayed for visual grading. For instance, for a MS Excel assignment where a graph is required and visual grading desired, the grading program will automatically open each file on right side of the screen, and allow typing of comments and points on a form on the left side of the screen.

The automated grading allows us to provide 24 hour turn-around on all assignments. This is a significant increase in turn-around over manual grading, providing increased contiguity. Students receive a response within 24 hours of working on the assignment. From the instructor perspective, this involves running the automated grader once per day (from personal experience over 40 spreadsheets submissions can be graded in less than 15 minutes).

One negative we found with the automated grader was that there may be “false positives” in identifying “wrong” answers. For instance, the grading template might specify the correct formula for a cell as =IF(B2>0,”Past Due”,”Paid in Full”). A student response of =IF (B2<=0,”Paid in Full”,”Past Due”) might or might not be correct, depending on the context. And there are many more variations that required the instructor to make a judgment, decreasing the benefits of automated grading. Currently the system permits the instructor to only build in two correct answers for each spreadsheet cell.

To address the rigidity of the automated grader, we allow students to submit assignments twice before the due date. Submissions are graded once per day, and students receive detailed feedback. For example, a MS Excel feedback message might be:

- 5 points, Incorrect formula in cell D17, annual interest rates must be divided by 12 for monthly payments, you had: PMT(.07,60,12000)

Of course, a typical submission might have many of these messages. Some of these messages will be “false positives”, meaning that a correct answer could be marked wrong because of a very strict grading template. Students are still required to make the change that satisfies the grading template, and submit the assignment again. We plan to increase the number of valid responses to reduce the number of false positives.

We have noticed several beneficial side effects to allowing re-submission of assignments. First, repetition is increased. Most students take advantage of re-submitting the assignment, so they revisit their work regularly. Learning is increased by having to fix errors in assignments, versus just going onto the next assignment. Second, students perceive more interaction and involvement in their assignment. Traditionally, students submit their work, and then receive it back after one or two classes have passed. In this scenario, their involvement ends at the time they hand it in. Many look at the bottom line score and never even read the comments. By submitting, then revising, the students actually read the comments, and revisit their own work. Another observation is that this approach of resubmission has worked very favorably in one section of the course that is entirely web based class. This is based on the increase of test scores after the resubmission of homework exercises was installed.

Final observations include that the grading has become much more objective for all assignments. The grading program can also be delegated to graduate assistants with complete confidence that the same grading scale an instructor is using is utilized by the graduate assistant. In addition we are able to consistently check many more formulas and values in a spreadsheet through the use of the automated grading feature versus manual checking of spreadsheets.

Items currently assessed

Currently the following items have been incorporated into the automatic grading system for spreadsheet applications:

- Name and existence of sheet
- Value of a particular cell (number as well as label)
- Formula in a particular cell
- Range name (cells associated with a range name)
- Use of range names in other formulas
- Capability to insert a value into one cell and see the impact on another cell (checking valid formulas)
- The ‘embedded’ ID of the student
- Graphs – existence
- Graphs – type
- Graphs – legend, axis labels
- Graphs – data range

One ongoing concern is that the many assignments could become merely busy-work that does not actually enhance learning. This can be a problem for over-burdened instructors; assignments that are easier

to grade can often become less valuable to the students. Furthermore, assignments that are lightly graded can deceive both instructors and students into thinking they have a better grasp of the material than they actually have.

As we have described above, the assignments that we assign are substantial and are graded in great detail by our system. Both students and instructors get accurate, detailed feedback to evaluate their grasp of the material. Informal feedback from the students indicates that they find the assignments challenging and worthwhile.

Overview of students and survey results

As noted earlier this automated grading feature was implemented in the same undergraduate course over 3 years in a total of six sections. Table 1 provides the demographics for the students enrolled in the class.

Table 1: Student demographics for the courses where the automated grading was implemented

Number of sections	6	
Average number of students/section	41	
Gender	Female	52.5
	Male	47.5
Class Year	Freshman	0.0
	Sophomore	20.5
	Junior	59.8
	Senior	18.9
Comfort level with personal computers at the beginning of the semester (compared to their peers)	High	21.3
	Above Avg.	42.6
	Average	32.8
	Below Avg.	2.9

Table 2 summarizes three questions from the university's standard student course evaluations as they relate to the automated grading. We are very encouraged by the end of the semester evaluations completed by the students in that we were not merely creating 'busy work' but student satisfaction has increased. The student course evaluations indicate that students perceive the work load to have decreased compared to other courses, while the feedback and fairness in grading scales increased.

Table 2: Student evaluations at the end of the semester (when compared to other business school courses)

	4 sections prior to implementation of automated grading	6 sections incorporating the automated grading
Amount of work	3.73	3.39
Feedback provided by instructor	3.91	4.21
Fairness in grading	3.23	3.95

Scale: 3.0 = equivalent to other courses, 4.0 =above average; 5.0 = significantly above average

Future Enhancements

Our plans for the future include: real-time grading, more grading abilities, and real-time assessment. All of these are feasible with today's technology, and are incremental changes to the solid core system that we have developed.

In order to increase the contiguity in the learning process, we would like for students to receive real-time feedback on submitted assignments. Currently, they have, at worst, 24 hours of delay between submitting the assignment, and receiving their grade and comments. We would like for them to submit the assignment, and on the next screen see their grade and comments. Their grades and comments would also be stored in the instructor's grade-book, and ideally, the student would receive their personalized feedback via email as soon as the project has been graded.

We have found that creating the text file grading templates can be time consuming and prone to errors. One of the enhancements we would like to include is a graphical user interface through which instructors could create templates by pointing-and-clicking, rather than text editing.

An additional enhancement is to provide more than two correct answers for spreadsheet formulas, permitting valid variations of the same answer. We envision an interactive grading program that would allow instructors to build grading templates on-the-fly, by identifying correct and incorrect answers in the files being graded. Instructors would only be queried about correctness of answers that were not in the "correct" and "incorrect" lists. As more correct and incorrect responses are identified, the instructor would be queried less and less, until assignments would be graded in a fully automatic mode.

Summary and Key Observations

The authors set out to reduce grading and administrative time through the use the automation techniques previously discussed. The current system has saved us time while improving the learning of the students. It allows us to require more assignments, grade them, provide detailed feedback, check for plagiarism, and increase the involvement of students in completing the assignments

Several key improvements to the courses resulted. One author was able to increase the number of assignments from five to twelve per semester without increasing the grading burden. The capability to pre-grade assignments and provide customized feedback has assisted students to look at their errors, correct them and learn from them, not merely look at the grade. Grading has become much more objective through the use of consistent templates.

In addition the assignments have become more customized per student reducing plagiarism. Through the use of embedding ID's and informing students that we would be checking for plagiarism our accuracy of plagiarism dropped from 12 students (out of 120 students) the first semester implemented to five in the most recent semester. The students are communicating that 'somehow' we are checking for duplicate work more closely.

Our experiments with automated grading and new class procedures enabled by emerging technologies have yielded significant results. We feel that the students are learning more because of our efforts. Positive informal student feedback, high faculty evaluations, and good test scores support our conclusions. The encouraging results motivate us to continue to explore the possibilities of automated grading, and technology-enhanced teaching.

References

- Anderson, J. and B. Reiser. 1985. "The LISP tutor." *Byte Magazine*, 10(4), 159-175.
- Gagne, R., Briggs, L., and W. Wager. 1988. *Principles of instructional design, 3rd edn.*, New York: Holt, Reinhard, & Winston.
- Hannafin, M. and K. Peck. 1988. *The Design Development and Evaluation of Instructional Software*. New York: MacMillian Publishing.

Murray, T. 1996. From story boards to knowledge bases, the first step in making CAI 'Intelligent'. In Carlson and Makedon (Eds.) *Educational Multimedia and Hypermedia*, Proceedings. (pp. 509-514). Charlottesville VA: AACE.

Saljo, R. 1979. Learning in the learner's perspective, some common sense conceptions. *Reports from the Institute of Education*, University of Gotherburg, 76.

Microsoft Office User Specialist (MOUS) Certification Requirements, <http://www.mous.net>, June 30, 2003.